k-Means Clustering with Distance-Based Privacy

Alessandro Epasto [*1] **Vahab Mirrokni** [*1] **Shyam Narayanan** [*2] **Peilin Zhong** [*1]

Abstract

In this paper, we initiate the study of Euclidean clustering with Distance-based differential privacy. Distance-based privacy is motivated by the fact that it is often only needed to protect the privacy of exact, rather than approximate, locations. We provide constant-approximate algorithms for k-means and k-median clustering, with additive error depending only on the attacker's precision bound ρ, rather than the radius Λ of the space. In addition, we empirically demonstrate that our algorithm performs significantly better than previous differentially private clustering algorithms, as well as naive distance-based private clustering baselines.

1. Introduction

Two of the most fundamental and widely studied problems in unsupervised machine learning are the k-means and k-median clustering problems. Solving these clustering problems can allow us to group together data efficiently, and hence extract valuable and concise information from massive datasets. The goal of the k-means (resp., k-median) clustering problem is: given a dataset X of points, construct a set C of k centers to minimize the clustering cost $\sum_{x \in X} d(x, C)^2$ (resp., $\sum_{x \in X} d(x, C)$), where $d(x, C)$ represents the minimum distance between the data point x and the closest center in C.

In general, machine learning and data mining algorithms are prone to leaking sensitive information about individuals who contribute data points. In certain scenarios, this can lead to severe consequences, including losses of billions of dollars (Neate, 2018) or even the loss of human lives (Baraniuk, 2015). Thus, providing accurate algorithms that protect data privacy has become crucial in algorithm design. Over the past decade, the notion of differential privacy (DP) (Dwork et al., 2006) has emerged as the gold standard for privacy-preserving algorithms, both in theory and in practice, and has been implemented by several major companies and the US Census (Erlingsson et al., 2014; Shankland, 2014; Ding et al., 2017; Abowd, 2018). Informally, DP requires the output (distribution) of the algorithm to remain almost the same under a small adversarial perturbation of the input. Hence, even the knowledge of all but one data point, along with the output of the algorithm, still cannot reveal significant information about the final data point.

The importance of k-means and k-median clustering, as well as preserving data privacy, has led to a large interest in designing differentially private clustering algorithms in Euclidean space (Blum et al., 2005; Nissim et al., 2007; Feldman et al., 2009; Gupta et al., 2010; Mohan et al., 2012; Wang et al., 2015; Nissim et al., 2016; Nock et al., 2016; Su et al., 2016; Feldman et al., 2017; Balcan et al., 2017; Nissim & Stemmer, 2018; Huang & Liu, 2018; Stemmer & Kaplan, 2018; Stemmer, 2020; Ghazi et al., 2020; Jones et al., 2021; Chang et al., 2021; Nguyen et al., 2021; Chaturvedi et al., 2022; Blocki et al., 2021; Cohen-Addad et al., 2022a; Epasto et al., 2022; Cohen-Addad et al., 2022b; Mahpud & Sheffet, 2022). Here, the goal is to design a differentially private set of k centers, such that the clustering cost with respect to these centers is only a small factor larger than the optimal (non-private) clustering cost. Importantly, the work of (Stemmer & Kaplan, 2018; Ghazi et al., 2020; Cohen-Addad et al., 2022b) led to efficient polynomial-time and differentially private algorithms that achieve constant multiplicative approximation ratios.

While we can obtain DP algorithms with low multiplicative error, all such algorithms also require an additional additive error. If Λ is the radius of a ball that is promised to contain all data points, even the best private clustering algorithms are known to have an additive error proportional to $\mathrm{poly}(k, d) \cdot \Lambda^p$, where $p = 2$ for k-means and $p = 1$ for k-median. This factor of Λ^p is in fact unavoidable, as a single individual datapoint can be moved up to distance Λ and the algorithm must preserve privacy with respect to this change. If we do not have a good bound of Λ, this factor may dominate the error, and may make the clustering algorithm highly inaccurate. Even if the bound is known exactly, errors scaling with Λ may however be unnecessary

[*] Equal contribution [1] Google Research [2] MIT. Correspondence to: Alessandro Epasto <aepasto@google.com>, Vahab Mirrokni <mirrokni@google.com>, Shyam Narayanan <shyamsn@mit.edu>, Peilin Zhong <peilinz@google.com>.

Presented at the 2^{nd} Annual Workshop on Topology, Algebra, and Geometry in Machine Learning (TAG-ML) at the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. 2023. Copyright 2023 by the author(s).

and unacceptable in certain situations.

The additive error depending on Λ is necessary because standard differential privacy requires us to protect learning *anything* about the location of any point. However, in practice this may not be necessary as it might be enough to not know the location of a point up to a certain error. For instance, in address data, the risk is leaking the actual location, but uncertainty within a few miles in a city is sufficient to protect the privacy of the person (Chatzikokolakis et al., 2013). Another motivation is in smart meters (Chatzikokolakis et al., 2013, Section 6.1), where accurately learning the fine-grained consumption can result in spectacular privacy leaks (e.g. learning which TV channel is being watched (Greveler et al., 2012; Lam et al., 2007)) but slight uncertainty on the measurements is sufficient to protect from such attacks. Moreover, when differential privacy is used to protect the algorithm from adversarial inputs, it is often sufficient to protect against small perturbations as large perturbations can be detected or removed otherwise (Lécuyer et al., 2019).

These cases can be modeled by variants of differential privacy, such as dX privacy (a.k.a. extended differential privacy) (Chatzikokolakis et al., 2013; Fernandes et al., 2021), and pixelDP (Lécuyer et al., 2019). All such models are adaptations or generalizations of DP which take into account a metric over the datasets.

In this paper, we study a concrete formulation of distance-based privacy which we call dist-DP. An algorithm is $(\varepsilon, \delta, \rho)$-dist-DP if the algorithm protects (ε, δ)-differential privacy of a single data point if it is moved by at most ρ in a metric space. This is a less restrictive version of DP, as usually the neighboring datasets are defined to be any two datasets with a single point allowed to move anywhere.

The main question we study in this paper is the following: can we obtain much better approximation results (and algorithms better in practice) if we allow the algorithm to resist small movements, as opposed to arbitrary movements, of a point for instance for clustering? In other words, can we design ρ-dist-DP algorithms that perform significantly better than the state of the art regular DP algorithms for k-means or k-median clustering?

1.1. Our Results

In this work, we answer the above question affirmatively, by providing an efficient and accurate theoretical algorithm, and showing empirically that our algorithm outperforms clustering algorithms with standard differential privacy.

1.1.1. Theoretical Results

From a theoretical perspective, we are able to obtain $O(1)$-approximate algorithms for k-means and k-median clustering with ρ-dist-DP, and with additive error essentially only depending on the smaller distance ρ as opposed to the full radius Λ. More precisely, our main theorem is the following.

Theorem 1.1. *Let n, k, d be integers, $\rho \in (0, \Lambda], \varepsilon, \delta \in (0, 1]$ be privacy parameters, and $p \in \{1, 2\}$. Then, given a dataset $X = \{x_1, \ldots, x_n\}$ of points in a given ball of radius Λ in Euclidean space $\mathbb{R}^d$, there exists a polynomial-time $(\varepsilon, \delta, \rho)$-dist-DP algorithm $\mathcal{A}$ that outputs a set of centers $C = \{c_1, \ldots, c_k\}$, such that*

$$\sum_{i=1}^{n} d(x_i, C)^p \le O(1) \cdot \min_{\substack{C^* \subset \mathbb{R}^d \\ |C^*|=k}} \sum_{i=1}^{n} d(x_i, C^*)^p + \mathrm{poly}\left(k, d, \log n, \frac{1}{\varepsilon}, \log\frac{1}{\delta}, \log\frac{\Lambda}{\rho}\right) \cdot \rho^p.$$

Here, $p = 1$ for k-median and $p = 2$ for k-means.

Qualitatively, Theorem 1.1 has similar guarantees to (Stemmer & Kaplan, 2018), who provided an (ε, δ)-differentially private algorithm with an $O(1)$-approximation algorithm and additive error that was $\mathrm{poly}(k, d, \log n, \frac{1}{\varepsilon}, \log\frac{1}{\delta}) \cdot \Lambda^p$. The main difference is that we drastically reduce the additive error by reducing the dependence on Λ to a dependence on the distance privacy parameter ρ.

Running time and parallel computation. The runtime of a straightforward implementation of our algorithm is $\tilde{O}(nkd) + \mathrm{poly}(k) \cdot d$,[1] if we also ignore polynomial factors in $\log\frac{\Lambda}{\rho}$. By using approximate near neighbor algorithms, we can improve this further to $\tilde{O}(nd) + \mathrm{poly}(k) \cdot d$, which for k at most a small polynomial in n, is nearly linear. In addition, the algorithm can be easily implemented in the massively parallel computation (MPC) model (Karloff et al., 2010; Beame et al., 2017) (an abstraction of MapReduce (Dean & Ghemawat, 2004)) using $O(1)$ rounds and near linear total space where each machine has sublinear space. We discuss these further at the end of Appendix C (see the Supplementary material).

Finally we remark that the ρ^p dependence in the additive error is required for ensuring ρ-dist-DP. We prove in Appendix D (see the Supplementary material) that any $(\varepsilon, \delta, \rho)$-dist-DP algorithm must incur $\Omega(k \cdot \rho^2)$-additive error for k-means and $\Omega(k \cdot \rho)$-additive error for k-median.

1.1.2. Empirical Results

We empirically studied the performance of our algorithm on public and real-world datasets. We compare the approximation guarantee of our algorithm with the standard DP clustering algorithm and the standard non-private k-clustering algorithm. Experiments show that our algorithm outperforms the DP clustering algorithm and is only slightly

[1] $\tilde{O}(f(n))$ denotes $O(f(n) \log f(n))$.

worse than the non-private algorithm. In addition, we show that smaller ρ provides a better approximation guarantee, which aligns with our theoretical study. We refer readers for more details of our empirical study to Section 3.

2. Technical Overview

In this section, we describe the high-level ideas for obtaining Theorem 1.1. For simplicity, in this overview we focus on k-median and assume the dimension is $d = (\log n)^{O(1)}$.

Our approach follows two high-level steps, inspired by the work of (Chen, 2009; Cohen-Addad et al., 2022b). The insight used in (Cohen-Addad et al., 2022b), which proved highly efficient private clustering algorithms, is to start by generating a crude but private solution that may use a large number of centers and have a large approximation, but has small additive error. Then, one can apply the crude solution to partition the Euclidean space $\mathbb{R}^d$ into smaller regions, and apply some regular differentially private clustering algorithm in the regions. We follow a similar high-level template to (Cohen-Addad et al., 2022b). However, we still need to implement each of these steps, which require several technical insights to ensure we maintain privacy while only losing additive error roughly proportional to $\mathrm{poly}(k, d) \cdot \rho$.

To obtain a crude approximation, we use a technique based on partitioning the space $\mathbb{R}^d$ into randomly shifted grids at various levels (also known as the Quadtree). In the Quadtree, the 0th level is a very coarse grid containing the large ball of radius Λ, and each subsequent level refines the previous level with smaller grid cells. For a single grid and knowledge of which point lies in which grid cell, a natural approach for minimizing cost would be to output the centers of the "heaviest" cells, i.e., those with the most number of points. Indeed, it is known that outputting the $O(k)$ heaviest cells at each grid level provides a good approximation, at the cost of having more than k centers.

While this is not DP, a natural way of ensuring privacy would be to add Laplace noise to each count and add the heaviest cells after this. Unfortunately, doing so will lead to error depending on the full radius Λ. For example, if there was only a single data point, there will be at least e^d cells even at coarse levels, and several of them may have large noisy counts. Hence, we are likely to choose completely random cells, which will cause additive error to behave like Λ as opposed to ρ. Another option is to add noise to the points first and then compute the heaviest cells. While this avoids additive dependence on Λ, the additive dependence will behave like $n \cdot \rho$ where n is the full size of the dataset.

Surprisingly, we show that we can *combine* both of these observations in the right way. Namely, for coarse cells (i.e., with length larger than $\tilde{O}(\rho)$), we add noise (of distance proportional to $\tilde{O}(\rho)$) to the data points directly to generate *private points* $\tilde{x}_i$, and then compute the heaviest cells without adding noise to the counts. For fine cells (length smaller than $\tilde{O}(\rho)$), we do not add noise to the data points, but we add Laplace noise to the cell counts.

To explain the intuition behind this, suppose that the n data points happen to be perfectly divided into n/k clusters, where every point has distance r to its nearest cluster center. If $r \gg \rho$, then even if we add $\tilde{O}(\rho)$ noise to each data point, we will still find cluster centers that are within $\tilde{O}(r)$ of each correct center. So, the k-means cost should only blow up by a small multiplicative factor, without additive error. Alternatively, if $r \ll \rho$, then the grid cells of side length $\tilde{O}(r)$ should contain the entire cluster, and hence have n/k points in them. Assuming $n \gg d \cdot k$, even if we add Laplace noise to each of e^d cells, none of them will exceed n/k. Alternatively, if $n \ll d \cdot k$, then our approach of simply adding noise to the points and obtaining $n \cdot \rho$ error will be only $O(dk) \cdot \rho$, which is small.

In summary, we can generate a crude approximation F with roughly $O(k)$ cells per grid level (and $\tilde{O}(k)$ centers total), with small additive ratio. But we desire for the number of centers to be exactly k, and the multiplicative ratio to be $O(1)$, whereas ours will end up being $d^{O(1)}$. To achieve such an accurate result, we use F to partition the data into regions, and apply a private coreset algorithm on each. By combining these coresets together, we may obtain a private coreset of the full data, and then we can apply an $O(1)$-approximate non-private algorithm on the coreset.

A first attempt, inspired by (Chen, 2009; Cohen-Addad et al., 2022b), is to send each x_i to a region S_j if $f_j \in F$ is the closest center to x_i, and then compute a standard (i.e., not dist-DP) private coreset on each region S_j. To avoid dealing with large additive errors depending on Λ, we further split each region into a close and far region, depending on whether the distance from x_i to f_j is more than or less than $S \cdot \rho$ for some parameter S.

This attempt will still suffer from a large additive cost. For instance, if a point moves, even by distance ρ, it may move from a close region to a far region. Hence, the far region may have 1 more point, and since the far regions have diameter Λ, an algorithm that is private to adding or deleting a point must incur error proportional to Λ.

Our fix for this is to assign each x_i to a region not based on its closest point and distance, but instead based on $\tilde{x}_i$'s closest point and distance, where we recall that $\tilde{x}_i$ the noisy version x_i. For the points $\{x_i\}$ that are mapped to a far region (meaning $\tilde{x}_i$ is far from its nearest f_j), we will simply use $\{\tilde{x}_i\}$ as the coreset, as $\tilde{x}_i$ is already dist-DP. However, for points that are mapped to a close region, while we use $\tilde{x}_i$ to determine which region the point x_i is mapped to, we compute a private coreset using (Stemmer & Kaplan, 2018) on the points x_i, rather than use the points $\tilde{x}_i$.

To explain why this algorithm is accurate, for the close regions, we obtain additive error proportional to $S \cdot \rho$ as we apply the private coreset on a ball of radius $S \cdot \rho$. There is one region for each center in F, which multiplies the additive error by $|F| = \tilde{O}(k)$. For the far regions, we first note that $d(\tilde{x}_i, C) = d(x_i, C) \pm \tilde{O}(\rho)$ for any set of k centers C, as $d(x_i, \tilde{x}_i) \le \tilde{O}(\rho)$. Hence, we have additive error $\tilde{O}(\rho)$ per point. While this seems bad as this might induce additive error for n points, we in fact show that this additive error can be "charged" to multiplicative error. To see why, if x_i mapped to the far regions, this means $d(\tilde{x}_i, F) \ge \rho \cdot S$, which also means $d(x_i, F) \ge \Omega(\rho \cdot S)$, If there were T such points, then the total cost of X with respect to F is at least $T \cdot \rho \cdot S$, whereas the additive error is roughly $T \cdot \rho$. Finally, in our crude approximation we show $\text{cost}(X; F)$ is at most $d^{O(1)}$ times the optimum k-means cost, which means for $S \gg d^{O(1)}$ the additive error is small even compared to the optimum cost. Hence, we can charge the additive error to multiplicative error. We still have additive error from the close regions, but for $S = d^{O(1)}$, the additive error is only $\text{poly}(k, d) \cdot \rho$.

To summarize, while our techniques are inspired by (Cohen-Addad et al., 2022b), one important novel technical contribution of our work is that while (Cohen-Addad et al., 2022b) uses the true locations of the points to assign them to regions, we first add Gaussian noise to the points to determine their region, and then use the noised points *only* for the "far" regions and the true points *only* for the "close" regions. This change is crucial in ensuring the analysis is successful. In addition, we must set several parameters carefully to charge the additional incurred cost either to a small additive or small multiplicative factor.

3. Empirical Evaluation

In this section, we study the emperical approximation of our ρ-dist-DP k-means clustering algorithm.

Datasets. We evaluate our algorithm on 6 well-known public datasets *brightkite* (51406×2), *gowalla* (107092×2), *shuttle* (58000×10), *skin* (Bhatt & Dhall, 2010) (245057×4), *rangequeries* (Savva et al., 2018) (200000×6) and *s-sets* (Fränti & Sieranoja, 2018) (5000×2). Brightkite and gowalla are datasets of geographic locations (latitude and longitude) of users and can be found in Stanford Large Network Dataset Collection (SNAP) (Leskovec & Krevl, 2014), shuttle, skin and rangequeries are non-geographic datasets and can be found on UCI Repository (Dheeru & Karra Taniskidou, 2017), and s-sets is another non-geographic dataset and can be found in the clustering benchmark dataset[2]. We preprocess each dataset to fit into $[-1, 1]^d$. We refer readers to Appendix E in the Supplementary material for more details of the preprocessing steps.

Setup. We compare our algorithm with three other algorithms. We report the k-means cost of all algorithms. The three compared baseline algorithms are as follows.

1. *Non-private baseline*: We compare our algorithm with the non-private k-means solver using k-means++ seeding implemeted by Python scikit-learn package (Pedregosa et al., 2011). The output k-means cost of this baseline can be regarded as the groudtruth cost.
2. *DP baseline*: This is a k-means clustering algorithm in the standard DP setting implemented in part of a standard open-source DP library [3].
3. *ρ-Dist-DP baseline:* We run non-private k-means solver on a dataset $\tilde{X}$, where we apply $(\varepsilon, \delta, \rho)$-dist-DP preserving noise directly to each data point.

In all experiments, we fix privacy parameters $\varepsilon = 1, \delta = 10^{-6}$. We evaluate our algorithms for different choices of the privacy parameter ρ. Note that the parameter ρ should not be determined by our algoirhtm. We try different ρ to show how the choice of ρ affects the clustering quality. We refer readers to Section 7 (in the Supplementary material) for more discussions of the choice of ρ.

Our Results. We run all algorithms for $k = 4, 6, 8, 12, 16$. For each experiment, we repeat 10 times and report the mean and the standard error. In the experiments shown in Figure 1 in the full paper (see the Supplementary material), we fix $\rho = 0.05$[4]. The k-means cost of our dist-DP k-means algorithm is always smaller than the cost of DP k-means baseline and is only slightly worse than the non-DP baseline which is as expected. The dist-DP baseline introduces a large k-means cost which implies that our partitioning straties are indeed necessary and can improve the clustering quality significantly in practice. Finally, we fix $k = 8$ and investigate how the changes of ρ affect the k-means cost of our dist-DP k-means algorithm. We run our algorithm on all datasets for $\rho = 1, 0.08, 0.008, 0.0001$. As shown in Figure 2 in the full paper, the k-means cost decreases as ρ decreases, which is as expected. For running time, though we did not optimize our implementation, each algorithm runs within at most a few minutes in a single thread mode.

In summary, for a reasonable range of ρ, we significantly outperform previous DP k-means algorithms, whereas more naive distance-based DP algorithms perform far worse. In addition, we have comparable approximation guarantees even to the non-private k-means algorithm.

[2] https://cs.joensuu.fi/sipu/datasets/.

[3] https://ai.googleblog.com/2021/10/practical-differentially-private.html.

[4] We show advantages of our clustering for an example ρ which neither depends on our algorithm nor be optimized. An example of the privacy guarantee of $\rho = 0.05$: For geographic (latitude and longitude) datasets (e.g., brightkite, gowalla), an attacker is hard to distinguish whether a user was in New York or in Toronto.

References

Abowd, J. M. The us census bureau adopts differential privacy. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 2867–2867, 2018.

Balcan, M.-F., Dick, T., Liang, Y., Mou, W., and Zhang, H. Differentially private clustering in high-dimensional euclidean spaces. In *International Conference on Machine Learning*, pp. 322–331. PMLR, 2017.

Baraniuk, C. Ashley madison:'suicides' over website hack. *BBC News*, 24, 2015.

Beame, P., Koutris, P., and Suciu, D. Communication steps for parallel query processing. *Journal of the ACM*, 64(6): 1–58, 2017.

Bhatt, R. and Dhall, A. Skin segmentation dataset. *UCI Machine Learning Repository*, 2010.

Blocki, J., Grigorescu, E., and Mukherjee, T. Differentially-private sublinear-time clustering. In *IEEE International Symposium on Information Theory (ISIT)*, pp. 332–337. IEEE, 2021.

Blum, A., Dwork, C., McSherry, F., and Nissim, K. Practical privacy: the sulq framework. In *Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of Database Systems (PODS)*, pp. 128–138, 2005.

Chang, A., Ghazi, B., Kumar, R., and Manurangsi, P. Locally private k-means in one round. In *International Conference on Machine Learning*, pp. 1441–1451. PMLR, 2021.

Chaturvedi, A., Jones, M., and Nguyen, H. L. Locally private k-means clustering with constant multiplicative approximation and near-optimal additive error. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 6167–6174. AAAI Press, 2022.

Chatzikokolakis, K., Andrés, M. E., Bordenabe, N. E., and Palamidessi, C. Broadening the scope of differential privacy using metrics. In *Privacy Enhancing Technologies - 13th International Symposium (PETS)*, volume 7981 of *Lecture Notes in Computer Science*, pp. 82–102. Springer, 2013.

Chen, K. On coresets for k-median and k-means clustering in metric and euclidean spaces and their applications. *SIAM J. Comput.*, 39(3):923–947, 2009.

Cohen-Addad, V., Epasto, A., Lattanzi, S., Mirrokni, V., Munoz, A., Saulpic, D., Schwiegelshohn, C., and Vassilvitskii, S. Scalable differentially private clustering via hierarchically separated trees. In *Knowledge Discovery and Data Mining (KDD)*, pp. 221–230, 2022a.

Cohen-Addad, V., Epasto, A., Mirrokni, V., Narayanan, S., and Zhong, P. Near-optimal private and scalable k-clustering. In *Advances in Neural Information Processing Systems*, 2022b.

Dean, J. and Ghemawat, S. Mapreduce: Simplified data processing on large clusters. 2004.

Dheeru, D. and Karra Taniskidou, E. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

Ding, B., Kulkarni, J., and Yekhanin, S. Collecting telemetry data privately. *Advances in Neural Information Processing Systems*, 30, 2017.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. D. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography, Third Theory of Cryptography Conference, (TCC)*, volume 3876 of *Lecture Notes in Computer Science*, pp. 265–284. Springer, 2006.

Epasto, A., Medina, A. M., Schwiegelshohn, C., Saulpic, D., Vassilvitskii, S., Lattanzi, S., Mirrokni, V., and Cohen-addad, V. P. Scalable differentially private clustering via hierarchically separated trees. In *Proceedings of the ACM SIGKDD international conference on Knowledge discovery and data mining*, 2022.

Erlingsson, Ú., Pihur, V., and Korolova, A. Rappor: Randomized aggregatable privacy-preserving ordinal response. In *Proceedings of the 2014 ACM SIGSAC conference on computer and communications security*, pp. 1054–1067, 2014.

Feldman, D., Fiat, A., Kaplan, H., and Nissim, K. Private coresets. In *Proceedings of the 41st Annual ACM Symposium on Theory of Computing (STOC)*, pp. 361–370. ACM, 2009.

Feldman, D., Xiang, C., Zhu, R., and Rus, D. Coresets for differentially private k-means clustering and applications to privacy in mobile sensor networks. In *2017 16th ACM/IEEE International Conference on Information Processing in Sensor Networks (IPSN)*, pp. 3–16. IEEE, 2017.

Fernandes, N., Kawamoto, Y., and Murakami, T. Locality sensitive hashing with extended differential privacy. In Bertino, E., Shulman, H., and Waidner, M. (eds.), *Computer Security - ESORICS 2021 - 26th European Symposium on Research in Computer Security, Darmstadt, Germany, October 4-8, 2021, Proceedings, Part II*, volume 12973 of *Lecture Notes in Computer Science*, pp. 563–583. Springer, 2021.

Fränti, P. and Sieranoja, S. K-means properties on six clustering benchmark datasets. *Applied intelligence*, 48:4743–4759, 2018.

Ghazi, B., Kumar, R., and Manurangsi, P. Differentially private clustering: Tight approximation ratios. In *Advances in Neural Information Processing Systems*, 2020.

Greveler, U., Glösekötterz, P., Justusy, B., and Loehr, D. Multimedia content identification through smart meter power usage profiles. In *Proceedings of the International Conference on Information and Knowledge Engineering (IKE)*, pp. 1. The Steering Committee of The World Congress in Computer Science, Computer . . . , 2012.

Gupta, A., Ligett, K., McSherry, F., Roth, A., and Talwar, K. Differentially private combinatorial optimization. In *Proceedings of the twenty-first annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 1106–1125. SIAM, 2010.

Huang, Z. and Liu, J. Optimal differentially private algorithms for k-means clustering. In *Proceedings of the 37th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pp. 395–408, 2018.

Jones, M., Nguyen, H. L., and Nguyen, T. D. Differentially private clustering via maximum coverage. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021.

Karloff, H., Suri, S., and Vassilvitskii, S. A model of computation for mapreduce. In *Proceedings of the twenty-first annual ACM-SIAM symposium on Discrete Algorithms (SODA)*, pp. 938–948. SIAM, 2010.

Lam, H. Y., Fung, G., and Lee, W. A novel method to construct taxonomy electrical appliances based on load signaturesof. *IEEE Transactions on Consumer Electronics*, 53(2):653–660, 2007.

Lécuyer, M., Atlidakis, V., Geambasu, R., Hsu, D., and Jana, S. Certified robustness to adversarial examples with differential privacy. In *2019 IEEE Symposium on Security and Privacy, SP 2019*, pp. 656–672. IEEE, 2019.

Leskovec, J. and Krevl, A. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June 2014.

Mahpud, B. and Sheffet, O. A differentially private linear-time fptas for the minimum enclosing ball problem. In *Advances in Neural Information Processing Systems*, 2022.

Mohan, P., Thakurta, A., Shi, E., Song, D., and Culler, D. Gupt: privacy preserving data analysis made easy. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pp. 349–360, 2012.

Neate, R. Over $119 bn wiped off facebook's market cap after growth shock. *The Guardian*, 26, 2018.

Nguyen, H. L., Chaturvedi, A., and Xu, E. Z. Differentially private k-means via exponential mechanism and max cover. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 9101–9108. AAAI Press, 2021.

Nissim, K. and Stemmer, U. Clustering algorithms for the centralized and local models. In *Algorithmic Learning Theory*, pp. 619–653. PMLR, 2018.

Nissim, K., Raskhodnikova, S., and Smith, A. Smooth sensitivity and sampling in private data analysis. In *Proceedings of the thirty-ninth annual ACM Symposium on Theory of computing (STOC)*, pp. 75–84, 2007.

Nissim, K., Stemmer, U., and Vadhan, S. Locating a small cluster privately. In *Proceedings of the 35th ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems*, pp. 413–427, 2016.

Nock, R., Canyasse, R., Boreli, R., and Nielsen, F. k-variates++: more pluses in the k-means++. In *International Conference on Machine Learning*, pp. 145–154. PMLR, 2016.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Savva, F., Anagnostopoulos, C., and Triantafillou, P. Explaining aggregates for exploratory analytics. In *2018 IEEE International Conference on Big Data (Big Data)*, pp. 478–487. IEEE, 2018.

Shankland, S. How google tricks itself to protect chrome user privacy. *CNET, October*, 2014.

Stemmer, U. Locally private k-means clustering. In *Symposium on Discrete Algorithms (SODA)*, pp. 548–559, 2020.

Stemmer, U. and Kaplan, H. Differentially private k-means with constant multiplicative error. In *Advances in Neural Information Processing Systems*, pp. 5436–5446, 2018.

Su, D., Cao, J., Li, N., Bertino, E., and Jin, H. Differentially private k-means clustering. In *Proceedings of the sixth ACM conference on data and application security and privacy*, pp. 26–37, 2016.

Wang, Y., Wang, Y.-X., and Singh, A. Differentially private subspace clustering. *Advances in Neural Information Processing Systems*, 28, 2015.